

WHAT IS CLAIMED IS:

1. A method of operating a computer system comprising:
determining if any performance state data stored in the computer system in a first area of memory specifying performance states for a plurality of processors is associated with a processor being utilized in the computer system; and
if a portion of the performance state data is found to be associated with the processor being utilized in the computer system, copying the portion of the performance state data into a second area of memory.
2. The method as recited in claim 1 where the performance state data is stored in performance state tables.
3. The method as recited in claim 1 wherein the method is performed in a portion of basic input output system (BIOS) and wherein the first area of memory is a memory segment utilized during power on self test (POST) processing by BIOS and wherein the second area of memory is a runtime memory segment of BIOS
4. The method as recited in claim 3, wherein the runtime memory segment starts at address 0xF000.
5. The method as recited in claim 1 wherein when the portion of the performance state data is found to be associated with the processor being utilized in the computer system, CPUID information associated with the portion of the performance state data matches CPUID information in the processor.
6. The method as recited in claim 1 wherein if a portion of the performance state data is not found to be associated with the processor being utilized in the computer system, determining if a near match exists between a portion of the performance state data and the processor being utilized in the computer system.

7. The method as recited in claim 1 further comprising generating performance state data if none of the performance state data is associated with the processor being utilized in the computer system and a near match does not exist.
8. A method of operating a computer system comprising:
evaluating information in a plurality of performance state tables, each of the performance state tables having information relating to a performance state of a processor; and
if a match is not found between one of the performance state tables and a processor being utilized in the computer system, determining if a near match exists and utilizing one of the performance state tables that is a near match to the processor if the near match exists.
9. The method as recited in claim 8, wherein determining if a near match exists comprises ignoring a stepping field in a CPUID field in the one of the performance state tables that is the near match to the processor.
10. The method as recited in claim 8, wherein the plurality of performance state tables are in a first area of memory, and if the near match is found, copying the information from the one of the performance state tables into a second area of memory, and wherein the first area of memory is a memory segment utilized during power on self test (POST) processing by basic input/output system (BIOS) and wherein the second area of memory is a runtime memory segment of the BIOS.
11. The method as recited in claim 8, further comprising :
if a match is not found one of the performance state tables and a processor being utilized in the computer system, and if the near match is determined not to exist, generating performance state data based on, at least in part, information provided in the processor.
12. A method of operating a computer system comprising:
determining if any performance state data stored in the computer system specifying performance states for a plurality of processors is associated with a processor being utilized in the computer system; and

generating performance state data if none of the performance state data is associated with the processor being utilized in the computer system.

13. The method as recited in claim 12, further comprising generating performance state data for the processor being used in the computer system after determining that a near match does not exist.

14. The method as recited in claim 12, wherein the generating the performance state data further comprises generating performance state information including a plurality of frequency values and a fixed voltage value.

15. The method as recited in claim 14, wherein an upper limit of frequency of the frequency values is determined according to providing an indication of maximum frequency in the processor.

16. The method as recited in claim 15, wherein the plurality of frequency values are spread approximately evenly between a lower limit and the upper limit.

17. A computer program product stored on computer readable medium operable in a computer system to:

- determine whether a match exists between a processor being utilized in the computer system and performance state information stored in a first area of memory; and
- if a match exists, copying matching performance state information into a second area of memory, wherein the first area of memory is a memory segment utilized during power on self test (POST) processing by basic input output system (BIOS) and wherein the second area of memory is a runtime memory segment of BIOS.

18. The computer program product as recited in claim 17, further operable to: generate a performance state data after no match is found to exist between any of the performance state data and the processor being utilized in the computer system.

19. The computer program product as recited in claim 18, further comprising generating performance state data for the processor being used in the computer system after determining that a near match does not exist.

20. The computer program product as recited in claim 18, wherein the generating the performance state data further comprises generating performance state table having a plurality of frequencies and in which processor voltage is fixed.

21. The computer program product as recited in claim 18, wherein an upper limit of the plurality of frequencies is determined in the generated performance state data according to an indication of maximum frequency in the processor and wherein the plurality of frequencies are spread approximately evenly between a lower limit and the upper limit.

22. The computer program product as recited in claim 17 wherein the computer program product comprises software operable to:

determine if a near match exists and utilize performance information from performance state data that constitutes a near match to the processor if the near match exists and if a match is not found between any of the performance data and the processor being utilized in the computer system.

23. A computer program product stored on computer readable medium operable in a computer system to:

determine whether a match exists between a processor being utilized in the computer system and any of a plurality of performance state data associated with respective processors; and
if a match is not found between any of the performance state tables and the processor being utilized in the computer system, the computer program product being operable to determine if a near match exists and utilize performance information from one of the performance state tables that is a near match to the processor if the near match exists.

24. The computer program product as recited in claim 23, wherein determining if a near match exists comprises ignoring a stepping field in a CPUID field and wherein if the near match exists, copying information from the near match data structure into the second memory area.

25. The computer program product as recited in claim 23, further comprising code operable to:

generate the performance state data based on information provided in the processor if a match is not found between any of the performance state tables and the processor being utilized in the computer system, and if the near match is determined not to exist.

26. A computer program product stored on computer readable medium operable in a computer system to:

determine whether a match exists between a processor being utilized in the computer system and performance state information for a plurality of processors; and
generate performance state data after no match is found to exist.

27. The computer program product as recited in claim 26, wherein the generating of the performance state data further comprises generating frequency performance state data in which processor voltage is fixed and frequency changes.

28. The computer program product as recited in claim 27, wherein an upper limit of frequency is determined in the generated performance state data according to an indication of maximum frequency in the processor.

29. The computer program product as recited in claim 28, wherein frequency values in the performance state data are spread approximately evenly between a lower limit and the upper limit.

30. An apparatus comprising:

means for determining if any performance state data specifying performance states for a plurality of processors, which is stored in a first area of

memory of a computer system utilized for system boot, is associated with a processor being utilized in the computer system; and means for copying a portion of the performance state data into a second area of memory if the portion of the performance state data is found to be associated with the processor being utilized in the computer system.

31. An apparatus comprising:

means for evaluating information in a plurality of performance state tables, each of the performance state tables having information relating to a performance state of a processor; and if a match is not found between one of the performance state tables and a processor being utilized in the computer system, means for determining if a near match exists and utilizing one of the performance state tables that is a near match to the processor if the near match exists.

32. An apparatus comprising:

means for determining if any performance state data stored in the computer system specifying performance states for a plurality of processors is associated with a processor being utilized in the computer system; and means for generating performance state data if none of the performance state data is associated with the processor being utilized in the computer system.

33. A computer program product stored on one or more computer readable media, operable in a computer system to:

copy one of a plurality of performance state data associated with respective processors from a disposable memory segment into a runtime memory segment; and generate an advanced configuration and power interface object based at least in part on the copied performance state data.

34. The computer program product of claim 33 wherein the advanced configuration and power interface object is statically generated.

35. The computer program product of claim 33 wherein the advanced configuration and power interface object is dynamically generated.

36. The computer program product of claim 33 wherein the first memory segment is utilized during power on self test (POST) processing by basic input/output system (BIOS), and wherein the second memory segment is a runtime memory segment of BIOS.

37. The computer program product of claim 33 wherein the copied performance state data has been determined to match a set of processor parameters that correspond to a processor in the computer system.

38. The computer program product as recited in claim 33, further operable to:
generate a performance state data after no match is found to exist between any
of the plurality of performance state data and the processor being
utilized in the computer system.

39. The computer program product as recited in claim 33, wherein the performance state data includes a plurality of frequencies and fixed processor voltage.

40. A computer system comprising:
memory that hosts a basic input/output system (BIOS) that selects a first of a
plurality of performance state information from a first area of memory
into a second area of memory, wherein the first area of memory is to
be utilized during power on self test (POST) processing by the BIOS,
and wherein the second area of memory is to be used for a runtime
memory segment of BIOS; and
a processor coupled with the memory.

41. The computer system of claim 40 wherein the first performance state information corresponds to a set of one or more processor parameters that correspond to the processor.

42. The computer system of claim 41 wherein the BIOS either selects the first performance state information as exactly matching the set of processor parameters or nearly matching.